Patent Claims:

1 – 24 (canceled)

25.   (new) A method for programming and/or executing programs for industrial automation systems, comprising:

providing a computer unit with input aids, output aids and a display device, having modules and functions respectively representing sub-tasks of an automation solution being modeled and/or created using the input aids and optionally the display device, having the modules and functions being structured and networked using the input aids and optionally the display device as to form a hierarchical tree as a machine-independent program,

wherein the machine-independent program is loaded in the form of a hierarchical tree into the corresponding components of the automation system and the corresponding components of the automation system execute the or each machine-independent program present in the form of at least one hierarchical tree.

26.   (new) The method according to claim 25, wherein the machine-independent program is executed on corresponding components of the automation system with the aid of at least one object machine assigned to the same.

27.   (new) The method according to claim 25, wherein the machine-independent program is present in the form of at least one hierarchical object or operator tree in the corresponding components of the automation system and is processed interpretatively.

28.   (new) The method according to claim 25, wherein the machine-independent program is present in the form of at least one object or operator tree with a structure equivalent or similar to the representation of the program in the or each display device.

29.   (new) The method according to claim 25, wherein the machine-independent program is loaded into the corresponding components of the automation system using a machine-independent, symbolic representation of the hierarchical tree.

30. (new) The method according to claim 29, wherein the machine-independent and symbolic representation of the hierarchical tree is in the form of a byte code or a markup language such as extended markup language.

31. (new) The method according to claim 25, wherein the object machine is configured as a real-time object machine with deterministic response and cycle times.

32. (new) The method according to claim 25, wherein the object machine provides operators and objects from which the machine-independent program is provided in the form of the hierarchical tree.

33. (new) The method according to claim 32, wherein the operators are instantiated during or after the loading of the machine-independent program and the symbolic representation of the hierarchical tree is converted to physical addresses to generate a loadable program.

34. (new) The method according to claim 25, wherein the object machine is implemented as a function unit that is closed and that processes the hierarchical tree to the runtime.

35. (new) The method according to claim 25, wherein the object machine is implemented in a distributed manner as at least one object, with the hierarchical object or operator tree processing itself.

36. (new) The method according to claim 25, wherein the modules and functions are assigned model information and/or meta-information using the input aids and/or the display device.

37. (new) The method according to claim 25, wherein the objects of the machine-independent program present as a hierarchical object or operator tree are assigned a collection of infrastructure services or infrastructure functions that access the objects or metadata assigned to the objects in a generic manner so that an infrastructure service or an infrastructure function can be used by all the objects and is applicable to all objects with metadata.

38.   (new) A device for programming and/or executing programs for industrial automation systems, comprising:

at least one computer unit with input aids, output aids and a display device;

a component for modeling and/or creating modules and functions, which respectively represent the sub-tasks of an automation solution;

a component for structuring the modules and functions and for networking the same, to form at least one hierarchical tree as at least one machine-independent program; and

a component to load the or each machine-independent program in the form of at least one hierarchical tree into the corresponding components of the automation system with the corresponding components of the automation system executing the machine-independent program present in the form of at least one hierarchical tree.

39.   (new) The device according to claim 38, wherein at least one object machine is assigned to the corresponding components of the automation system to execute the machine-independent program.

40.   (new) The device according to claim 38, wherein the machine-independent program is present in the form of at least one object or operator tree with a structure equivalent or similar to the representation of the program in the display device.

41.   (new) The device according to claim 39, wherein the object machine is configured as a real-time object machine with deterministic response and cycle times.

42.   (new) The device according to claim 38, wherein the object machine provides operators, and objects, from which the or each machine-independent program is provided in the form of the or each hierarchical tree.

43.   (new) The device according to claim 38, further comprising a device for assigning model information and/or meta-information to the modules and functions.

44.   (new) The device according to claim 38, wherein the object machine is implemented as a function unit that is closed and processes the or each hierarchical tree to the runtime.

45.　(new) The device according to claim 38, wherein the object machine is implemented in a distributed manner as at least one object, with the hierarchical object or operator tree processing itself.

46.　(new) The device according to claim 38, wherein the objects of the machine-independent program present as a hierarchical object or operator tree are assigned a collection of infrastructure services or infrastructure functions that access the objects via the containers assigned to the objects so that an infrastructure service or infrastructure function can be used by all the objects.

47.　(new) A Computer program implementing a method, comprising:

providing a computer unit with input aids, output aids and a display device, having modules and functions respectively representing sub-tasks of an automation solution being modeled and/or created using the input aids and optionally the display device, having the modules and functions being structured and networked using the input aids and optionally the display device as to form a hierarchical tree as a machine-independent program,

wherein the machine-independent program is loaded in the form of a hierarchical tree into the corresponding components of the automation system and the corresponding components of the automation system execute the or each machine-independent program present in the form of at least one hierarchical tree.